Developing A System Reliability Model Structure for Complex Systems: Challenges and Progress at Developing Improved Tools

Andrew Koehler
Statistical Sciences
Los Alamos National Laboratory
Los Alamos, New Mexico
USA
akoehler@lanl.gov

Richard Klamann
Statistical Sciences
Los Alamos National Laboratory
Los Alamos, New Mexico
USA
rmk@lanl.gov

Abstract

GROMIT (Graphical Representation Ontology Modeling Inference Tool) is a software package developed by the Statistical Sciences Group (D-1) at Los Alamos National Laboratory. It is part of the System Ethnography and Qualitative Modeling (SEQM) team's effort to advance research in socio-technical systems representations and system statistical reliability analysis. GROMIT supports system analysis by providing a robust, compact and dynamic graphical language to describe complex system structure. GROMIT forces a consistent integration of information on component composition with behaviors and then uses this rigorous foundation to infer system-wide behaviors from observed and/or elicited data.

1. Why is GROMIT being developed?

GROMIT supports the development of statistical reliability models of complex systems for which no single individual has a complete understanding. It does this by:

a) Capturing hypotheses from all system stakeholders about what components exist in the system, and how those components relate to one another;
b) Encoding component behaviors as set of rules which can tested against observed system behaviours;
c) Incorporating dynamic system behaviors across all operational modes of the system;
d) Linking component state information to quantitative and qualitative data sources;
e) Performing checks to determine whether component reliability hypotheses are consistent and result in calculable reliability models;
f) And inferring all possible combinations of component states that can result in observed system behaviors.

Our initial thrust is to describe the logic and structure of statistical system models by making use of all available system data, whether qualitative or quantitative. However, the longer term goal of the SEQM team is to advance the ability of planners to successfully deploy and operate complex systems within culturally, physically and politically defined constraints.

2. Development of a rigorous, compact representational scheme for system description

A key challenge of the GROMIT project has been to develop an effective visual schema for system representation. At first glance this may seem a trivial problem; many system visual description schemas already exist from a host of disciplines involved in system design and operation. However, in order to translate qualitative system design and functional logic into primarily quantitative statistical system descriptions, GROMIT needs to support a visual system language with a unique set of properties:

1. The visual language must be compact. Prior system analysis work had resulted in diagrams involving thousands of distinct entities—resulting in diagrams far too large to be conceptually manageable (on the order of 900 square feet in one case).

2. The visual language needs to be based off a set of flexibly definable ontologies. Because of the tremendous proliferation of components and diagrams that need to be described as part of a complex system, a tool was needed that could track system descriptions and components across dozens of representations.

3. The visual language needs to facilitate a "composable" view of system behavioral description. The SEQM effort is not a systems engineering effort -- reliability analyses involve systems that either already exist, or have already reached relatively mature design stage. Accordingly, our main problem is to build up a consistent system structural model based on widely different perspectives of the system, rather than determining how user requirements are best translated to system features.

4. The visual language has to be relevant across disciplines. While trying to retain the expressive flexibility necessary to describe qualitative system operation, a system visual language is needed that can communicate effectively with engineers, social scientists, and statisticians.

5. The language must be capable of describing dynamic concepts. System behavior is a combination of component, linkages, states, temporal, and conditional logics. The results of these interactions cannot be described in a manageable set of static diagrams for any but the most trivial of systems. Accordingly, the visual language used by GROMIT is designed for automation.

3. The Process of Using GROMIT in Support of Systems Reliability Modelling

In basic form, GROMIT starts with an attempt to derive a basic hypothesis for how an engineered components and system behavior interrelate using all available information sources. Expert elicitation is supported through the development of a common taxonomy for components and their relationships. As these taxonomies develop, GROMIT creates multiple interrelated ontology structures to help users track specific concepts, as well as to present information based on whatever categories and naming scheme are locally relevant. Entities, frames, ICOMS, channels, states, activities and events are all managed through these structured lists. Because GROMIT is used to reconcile multiple perspectives on system composition, with some exceptions, these lists are highly customizable and are based on lattice structures rather rigid inheritance or object oriented notions of organization.

As system information ontologies develop, GROMIT allows users to populate objects in the system with information about data sources, variable types, notes, and other kinds of contextual knowledge useful for system description and creation of a statistical model. It is anticipated that as GROMIT's ontology features develop, the tool may also strongly support information archives associated with analysis efforts. While documenting the sources and types of relationships that subject matter experts use to describe how the system performs (or fails to perform) its functions, GROMIT is capable of maintaining multiple system views and checking the consistency of those views as part of helping the elicitation team negotiate with the system participants a commonly agreed upon system understanding. These hypotheses are managed through a constantly evolving set of graphical system representations, using SEQML. Additionally, GROMIT has been designed to assist in analysis of systems (common in the case of weapons) which may be produced in many different variants, or which may be modified after production in different ways between units in the same stockpile population.

Once a common system form(s) have been checked for consistency, and agreed to by system participants, entities can be tied to available data sets through the use of links between components and datasets created in entity description fields. Additionally, entity state values can be coordinated with the values appearing in these linked datasets so that GROMIT's logical checks and scenario developments can be checked against the patterns of entity diagnostics observed in the dataset. This process helps to develop the structure of an overall system behavioral time-line, so that dynamic and causal parsing out of component combinations can occur. Through the addition of a time vector to component-system relationships, it is also possible to limit the degree to which consistency analyses must parse through all possible combinations, accordingly this reduces a great many possible component combinations from further analysis.

Entity behaviors are linked to temporal and system behaviors through the use of a GUI based conditional programming capability in GROMIT. Through a set of simple "if…then…else" statements users program how behaviors change based on the system activity, and ICOM links between entities. Through forward chaining performed by GROMIT it is possibly to verify that system behaviors are being properly described through the composition of entity behaviors established in the system. As a result, GROMIT allows very complex interactions and feed-back effects that are vital for system descriptions to be created through a gradual assembly process as system understanding becomes robust.

The results from linking entity behavioral logic into a temporal structure of activities and events, allows the GROMIT system model to generate all of the legal combinations of entity states compatible with the behaviors which have been described. In SEQML, these combinations of entity states are called "scenarios." Usually, scenarios will be automatically generated by GROMIT; they follow from the logical implications of the individual ICOM, event, and state programs associated with each entity. As part of describing how entities behave as part of the system's operational possibilities, users are in fact producing a great many descriptions of the combination of state and ICOMs which may be compatible with particular events.

3.1 Scenarios

GROMIT uses an advance set of backwards and forwards chaining methods, tied to specific activity and event logics, to assist the user create a robust behavioral scenario list based on the composition logic described by system participants. In addition to using automatically generated backward-chained scenarios, users also have the ability to test system hypotheses using forward-chained logic to test the implications of setting entities in particular states, or by changing particular ICOMS.

Scenarios serve three purposes. First, scenarios allow users to check how well they understand the dynamics of their system by comparing scenarios to telemetry or other entity datasets. If SEQML models are robust, a good overlap should exist between the combination of entity states observed in functional testing and those generated by GROMIT's logic rules. Second, scenarios assist in diagnosing entity reliability and behaviors based on what features of the system are measured. Finally, scenarios help check the implication of entity rules against natural biases in human cognition of complex systems, such as a tendency to too quickly diagnose single root causes for failures (versus with-holding judgment to consider all ways a particular event could be caused) as well as limits on abilities to understand complex dynamics.

With a consistent and calculable system representation, which are checked through the creation of scenarios, the structure of a validated reliability model may then be extracted semi-automatically from GROMIT and put into a formal statistical model of the system. GROMIT in particular is intended to generate MCMC model structure files compatible with either WinBugs or YADAS (a custom tool developed by the Statistical Sciences group at Los Alamos National Laboratory). We also foresee using the capabilities of GROMIT to explore other issues associated with complex system development, besides statistical reliability modeling. For example, the ability to trace back along system component pathways to develop all possible "legal" component combination alternatives provides a natural means by which to explore issues of system complexity from the standpoint of organizational management capabilities. Likewise, we also foresee use of the tool in a variety of system forensics applications. This forensics role will be described in more detail in the following section on future development plans for GROMIT.

References

Bernus, Peter, Kai Mertins, and Gunter Schmidt, eds. *Handbook on Architectures of Information Systems, International Handbooks on Information Systems*. New York, NY: Springer-Verlag, 2003.

Callon, Michel, John Law, and Arie Rip, eds. *Mapping the Dynamics of Science and Technology, Sociology of Science in the Real World*. London, UK: The Macmillan Press, Ltd., 1986.

Dorner, Dietrich. *The Logic of Failure*. Reading, MA: Addison-Wesley, 1996.

Forsberg, Kevin, Hal Mooz, and Howard Cotterman. *Visualizing Project Management*. New York, NY: John Wiley & Sons, Inc., 1996.

J. Langan-Fox, A. Wirth, S. Code, K. Langfield-Smith, A. Wirth. "Analyzing Shared and Team Mental Models." *International Journal of Industrial Ergonomics* 28 (2001): 99-112.

Kahneman, Amos Tversky and Daniel. "Extensional Versus Intuitive Reasoning: The Conjunction Fallacy in Probability Judgment." *Psychological Review*, 06/26/2003 1983, 293-315.

Knowledge Elicitation: Principles, Techniques and Applications. Edited by Dan Diaper. 1 ed. 1 vols. Vol. 1, *Expert Systems*. Liverpool: Ellis Horwood Limited, 1989. Reprint, Halsted Press.

Masuch, Michael. *Knowledge Representation and Reasoning under Uncertainty*. Edited by Laszlo Polos. Vol. 808, *Lecture Notes in Artificial Intelligence*. Berlin: Springer-Verlag Berlin Heidelberg, 1994.

Mathieson, Kieran. "Reducing Bias in Users' Evaluations of Information Systems." *Information & Management* 25 (1993): 165-71.

Perrow, Charles. *Normal Accidents*. New York, NY: Basic Books, Inc., 1984.

Pinch, Trevor. "How Do We Treat Technical Uncertainty in Systems Failure? The Case of the Space Shuttle Challenger." In *Social Responses to Large Technical Systems*, edited by Todd Laporte, 143-57. Boston: Kluwer Academic Publishers, 1991.

Quintas, Paul, ed. *Social Dimensions of Systems Engineering, Ellis Horwood Series in Interactive Information Systems*. London, UK: Ellis Horwood, 1993.

R. Hoffman, N. Shadbolt, A Burton, and G. Klein. "Eliciting Knowledge from Experts: A Methodological Analysis." *Organizational Behavior and Human Decision Processes* 62, no. 2 (1995): 129-58.

Randall Davis, Howard Shrobe, Peter Szolovits. "What Is Knowledge Representation?" *AI Magazine* 14, no. 1 (1993): 17-33.

Richardson, George P. "Problems with Causal-Loop Diagrams." *System Dynamics Review* 2, no. Summer (1986): 158-70.

US Air Force Systems Command. "MIL-STD-499A." Washington, DC: USAF, 1974.